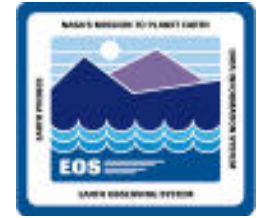# End to End Performance Analysis

Mary Armstrong

marmstro@eos.hitc.com

2 November 1995

Overview

End- to-end performance allocation is needed to

- **allocate cross-subsystem performance requirements to specific hardware, providing performance envelopes for sizing and testing;**

- **provide assurance that impacts of simultaneous system operations are taken into account in hardware sizing analysis**

- **examine anomalous conditions**

This presentation addresses how end to end performance will be analyzed.

Approach

- **Build "bottom-up" from subsystems, using**
  - **benchmarks to measure performance of specific functions**
  - **analysis to estimate performance of databases with respect to ECS transaction types**
  - **simulation/dynamic modeling to estimate complex processing functions and processing/disk interactions**

- **Develop scenarios to examine maximum loading conditions (e.g., 2X distribution during peak processing period) across the system**

- **For each system component, analyse the sum of all processing loads to develop performance envelopes**

- **Iterate based on software and hardware design changes, prototype results, and changes in expected loading**

Sizing Subsystem Components

Performance Measurement

- CSS DCE-based services
- FDDI switch/router
- Sybase, Illustra basic transactions
- Robot
- MSS COTS

Performance Analysis

- Sybase transactions within DM, DSS, Planning
- Illustra transactions within DSS

Performance Simulation

- Science Production (CPU, I/O, disk)
- Data Server (disk, I/O, robots)
- Networks (processing to data server network utilization)
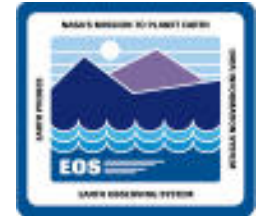
Push/Pull Analysis

Pull Analysis

- **Map benchmark results to ECS functionality (e.g., Illustra spatial search results mapped to data server services) to obtain values for "ECS transactions"**

- **Map transactions to hardware components**

- **Apply loads via user model, given derived or assumed values for**
  - **# service invocations per DAAC per hour**
  - **size of request (data volume requested)**

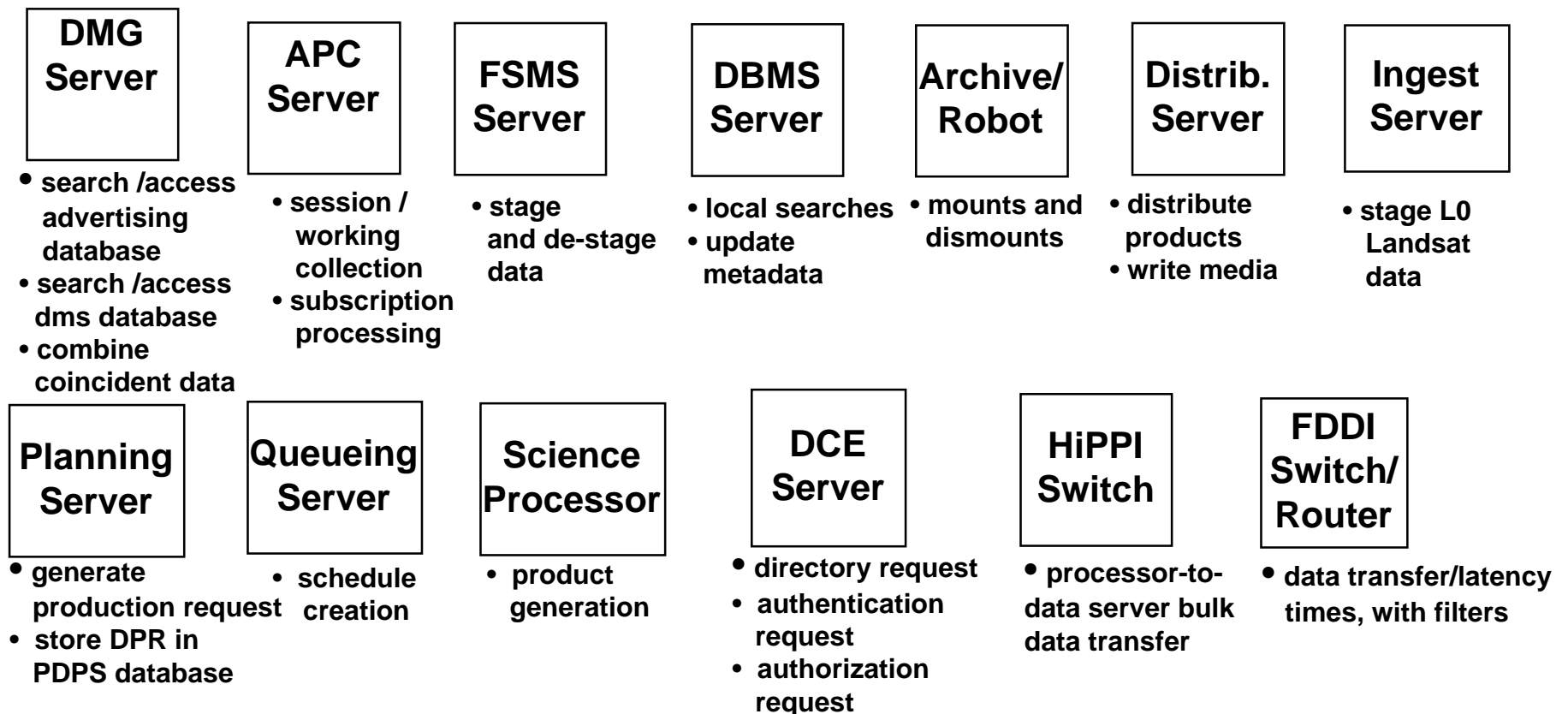- **Add overhead for infrastructure (CSS, MSS, O/S)**

Push Analysis

- **Develop push models (static and dynamic) to reflect peak and nominal resource utilization**

- **Perform sensitivity analysis on areas of uncertainty, including**
  - **reprocessing**
  - **processing on demand and subsetting**
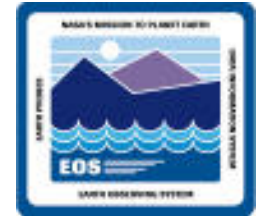
- **Add overhead for infrastructure (CSS, MSS, O/S)**

Scenario Example

1. Coincident Search
2. DAR for on-demand process (ASTER)
3. Ingest of Landsat
4. MODIS Production
5. Large Distribution

DMG Server
- search /access advertising database
- search /access dms database
- combine coincident data

APC Server
- session / working collection
- subscription processing

FSMS Server
- stage and de-stage data

DBMS Server
- local searches
- update metadata

Archive/ Robot
- mounts and dismounts

Distrib. Server
- distribute products
- write media

Ingest Server
- stage L0 Landsat data

Planning Server
- generate production request
- store DPR in PDPS database

Queueing Server
- schedule creation

Science Processor
- product generation

DCE Server
- directory request
- authentication request
- authorization request

HiPPI Switch
- processor-to-data server bulk data transfer

FDDI Switch/ Router
- data transfer/latency times, with filters

* Note that DCE client and MSS logging transactions should be added to all processes.

Next Steps/Plan for CDR

- **Current status**
  - Preliminary performance allocations have been made, based on benchmark and analysis results
  - Performance allocations do not take into account many new Release B services and COTs selections

- **Next steps**
  - Develop end-to-end scenarios for model implementation
  - Perform additional analyses and benchmarking (e.g., Release B COTs, Sybase, Illustra new features)
  - Iterate based on results, scenario variations, and different H/W design candidates